A Novel Framework for Incorporating Labeled Examples into Anomaly Detection

Jing Gao* Haibin Cheng† Pang-Ning Tan ‡

Abstract

This paper presents a principled approach for incorporating labeled examples into an anomaly detection task. We demonstrate that, with the addition of labeled examples, the anomaly detection algorithm can be guided to develop better models of the normal and abnormal behavior of the data, thus improving the detection rate and reducing the false alarm rate of the algorithm. A framework based on the finite mixture model is introduced to model the data as well as the constraints imposed by the labeled examples. Empirical studies conducted on real data sets show that significant improvements in detection rate and false alarm rate are achieved using our proposed framework.

1 Introduction

Anomalies or outliers are aberrant observations whose characteristics deviate significantly from the majority of the data. Anomaly detection has huge potential benefits in a variety of applications, including the detection of credit card frauds, security breaches, network intrusions, or failures in mechanical structures.

Over the years, many innovative anomaly detection algorithms have been developed, including statistical-based, depth-based, distance-based, and density-based algorithms [6, 2, 3]. While these algorithms prove to be effective in detecting outliers in many datasets, they may not be able to detect some outliers that are difficult to identify. Let's examine some scenarios where outliers are quite hard to find. In some data sets, the data used to create a profile of the "normal" behavior may not be representative of the overall population. As a consequence, it is not easy to distinguish between true outliers and previously unseen normal observations. Furthermore, although anomalies are, by definition, relatively rare, their number of occurrences may not be infrequent in absolute terms. For example, in intrusion detection, the number of network connections associated with an attack such as denial-of-service can be quite large. Such attacks are usually quite hard to detect.

To further illustrate this problem, consider the synthetic dataset shown in Figure 1. The normal points are represented as dots, while the outliers are represented as plus signs. The normal points are generated using two Gaussian distributions, centered at (15,20) and (70,90), respectively. The outliers are concentrated at the bottom right-hand corner of the diagram. Without labeled information, it is difficult to distinguish the true outliers from the low density normal points at the bottom left-hand corner. As a result, many existing anomaly detection algorithms tend to perform poorly on such data sets—some algorithms would consider all the low density normal points to be outliers, while others would consider the outliers to be normal. If labeled examples for some of these normal and outlying points are available, we may discriminate the true outliers from low-density normal points more effectively. This is the underlying strategy adopted by our proposed semi-supervised anomaly detection algorithm.

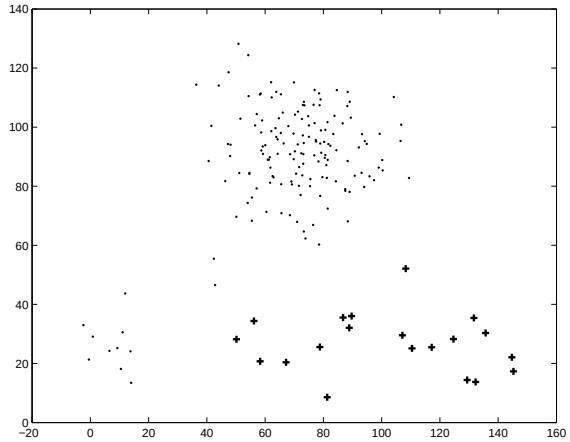

Figure 1: Example of a 2-D Data Set

In this work, we propose a novel probability-based approach for incorporating labeled examples into the anomaly detection task. A framework based on finite mixture model is introduced to model the data as well as the constraints imposed by the labeled examples. Anomalies are found by solving a constrained optimization problem using the standard Expectation-Maximization (EM) algorithm. Two variants of the algorithm are considered, namely, hard and soft (fuzzy) detection algorithms. Experimental results using real

*Michigan State University. Email: gaojing2@cse.msu.edu
†Michigan State University. Email: chenghai@cse.msu.edu
‡Michigan State University. Email: ptan@cse.msu.edu

data sets show that our proposed algorithm outperform unsupervised anomaly detection algorithms (including distance-based and density-based algorithms). We also show that our semi-supervised anomaly detection algorithm works better than semi-supervised classification algorithm for data sets with skewed class distribution.

2 Methodology

Let $X = \{x_1, x_2, \cdots, x_N\}$ denote a set of N data points drawn from a d-dimensional space, R^d. Following the approach in [3], we assume each data point has a probability π_0 of being an outlier and a probability $(1 - \pi_0)$ of being normal. π_0 is usually chosen to be a small number between 0 and 1. We further assume that the normal points are generated from a gaussian mixture model of k components whereas the outliers come from a uniform distribution.

Let $T = [t_{ij}]$ denote an $N \times k$ configuration matrix, where $t_{ij} = 1$ if x_i belong to the j-th mixture component. If $\sum_{j=1}^{k} t_{ij} = 0$, then x_i is considered an outlier. Assuming that the data points are independent and identically distributed, the conditional distribution of the dataset is:

$$P(X|T) = \left[(1-\pi_0)^{|M|} \prod_{x_i \in M} P_M(x_i|t_i)\right] \times \left[\pi_0^{|U|} \prod_{x_j \in U} P_U(x_j|t_j)\right] \tag{2.1}$$

Since the outliers are uniformly distributed, $P_U(x_j|t_j) = 1/Z_0$ where Z_0 is a constant. Furthermore, $|M| = \sum_{i,j} t_{ij}$ and $|U| = N - \sum_{i,j} t_{ij}$. The conditional distribution $P_M(x_i|t_i)$ is given by a mixture of normal distributions:

$$P_M(x_i|t_i) = \prod_{j=1} \left[\alpha_j p(x_i|\theta_j)\right]^{\sum_j t_{ij}}$$

where $\alpha_h \geq 0$ $(h = 1, \cdots, k)$, $\sum_{h=1}^{k} \alpha_h = 1$. Each mixture component is normally distributed:

$$p(x_i|\theta_j) = \frac{1}{(2\pi)^{d/2}|\Sigma_j|^{1/2}} \exp[-\frac{1}{2}(x_i - \mu_j)^t \Sigma_j^{-1}(x_i - \mu_j)]$$

where μ_j and Σ_j are the corresponding mean and covariance matrix associated with the parameter vector θ_j. Following the approach taken by Mitchell in [7], the model can be further simplified by assuming that each of the k mixture components is equally probable; i.e., $\forall j : \alpha_j = 1/k$.

Labeled examples impose additional constraints on how the configuration matrix T should be determined. Following the strategy used by Basu et al. [1], the constraints provided by labeled examples are modeled using Hidden Markov Random fields (HMRF). If M is the set of known outliers mislabeled as normal points and U is the set of known normal points mislabeled as outliers, then the prior probability of a configuration matrix T can be expressed as follows:

$$P(T) = \frac{1}{Z_2} \exp\left[-C_{10} \sum_{x_i \in M} (1 - \sum_{j=1}^{k} t_{ij}) - C_{01} \sum_{x_i \in U} \sum_{j=1}^{k} t_{ij}\right], \tag{2.2}$$

where C_{01} and C_{10} are the respective costs of misclassifying normal points as outliers and outliers as normal points. Intuitively, Equation 2.2 gives a lower probability to a configuration matrix that misclassifies many labeled examples.

Our objective is to maximize the following posterior probability:

$$P(T|X) = \frac{1}{Z_3} P(T) P(X|T) \tag{2.3}$$

Taking a logarithm on Equation 2.3 and substituting Equations 2.1 and 2.2 into the formula lead to the following objective function to be minimized by our semi-supervised anomaly detection algorithm:

$$\begin{aligned} Q &= \sum_{i=1}^{N} \sum_{j=1}^{k} t_{ij} D_{ij} + \gamma \sum_{i=1}^{N} (1 - \sum_{j=1}^{k} t_{ij}) \\ &+ C_{10} \sum_{x_i \in M} (1 - \sum_{j=1}^{k} t_{ij}) + C_{01} \sum_{x_i \in U} \sum_{j=1}^{k} t_{ij} \end{aligned} \tag{2.4}$$

where γ is a constant and

$$D_{ij} = \frac{1}{2}\left[(x_i - \mu_j)^T \Sigma_j^{-1}(x_i - \mu_j) - \log|\Sigma_j|\right].$$

3 Algorithms

This section describes our proposed algorithm based on the EM framework for solving the constrained optimization problem given in Equation 2.4.

3.1 EM Framework We first randomly initialize the parameters for μ_j and Σ_j $(j = 1, 2, \cdots, k)$. During the E-step, we determine the configuration matrix T by assigning each data point either to the outlier component or one of the k Gaussian components in such a way that minimizes the optimization function given in Equation 2.4. During the M-step, the parameters μ_j and Σ_j are re-estimated based on the current configuration matrix T.

There are two strategies for assigning data points to the outlier or Gaussian mixture components. The first approach, which we called *hard* anomaly detection, assigns a data point to only one of the $k + 1$ components. The second approach, which we called *soft* or *fuzzy* anomaly detection, assigns a data point either to the outlier component or to every Gaussian component with varying degrees of membership.

3.2 Hard Detection In hard detection, the configuration matrix T is a binary 0/1 matrix that satisfies the following

constraints: $t_{ij} \in \{0,1\}, \sum_{j=1}^{k} t_{ij} \leq 1$ $(1 \leq i \leq n, 1 \leq j \leq k)$. In order to minimize Q, we should minimize the contribution of every point to the overall objective function. For each point x_i, we can minimize the first term in Q by assigning x_i to the j^*-th component such that $j^* = \text{argmin}_j D_{ij}$. After the initial assignment, every point is now considered to be normal, i.e., $\forall i : \sum_{j=1}^{k} t_{ij} = 1$. To improve the assignment, we evaluate the possibility of relabeling some of the points as outliers, i.e. converting t_{ij^*} from one to zero. To do this, we first express the change in objective function as a result of relabeling as a matrix equation, $Q' = A^T T^* + b$, where $T^* = \{t_{ij^*}\}_{i=1}^{N}$ is the current configuration matrix, $A = \{a_i\}_{i=1}^{N}$ is a vector of coefficients expressed in terms of D_{ij}, γ, C_{10}, and C_{01}, while b is a constant that does not affect the label assignment. Note that optimizing Q' is equivalent to a linear programming problem. If $a_i > 0$, setting t_{ij^*} to zero will minimize its contribution to the objective function [9]. On the other hand, if $a_i \leq 0$, t_{ij^*} should retain its previous value of 1 in order to minimize Q'.

During the M-step, upon differentiating Q with respect to μ_j and Σ_j and equating them to zero, we obtain the following update formula for the mean vector and covariance matrix of each component:

$$\mu_j = \frac{\sum_{i=1}^{N} t_{ij} x_i}{\sum_{i=1}^{N} t_{ij}}$$

$$\Sigma_j = \frac{\sum_{i=1}^{N} t_{ij} (x_i - \mu_j)^T (x_i - \mu_j)}{\sum_{i=1}^{N} t_{ij}}$$

3.3 Fuzzy Detection Fuzzy clustering has been extensively researched over the years [5]. Instead of assigning a data point to a single cluster, each point belongs to a cluster with certain degree of membership. For fuzzy detection, the objective function is modified as follows:

$$\begin{aligned} Q &= \sum_{i=1}^{N} \sum_{j=1}^{k} (t_{ij})^m D_{ij} + \gamma \sum_{i=1}^{N} \left[1 - (\sum_{j=1}^{k} t_{ij})^m \right] \\ &+ C_{10} \sum_{x_i \in M} \left[1 - (\sum_{j=1}^{k} t_{ij})^m \right] \\ &+ C_{01} \sum_{x_i \in U} \left[\sum_{j=1}^{k} t_{ij} \right]^m \end{aligned} \tag{3.5}$$

where $m > 1$ is called the fuzzifier. Here, the configuration matrix T satisfies the constraints: $0 \leq t_{ij} \leq 1$, $\sum_{j=1}^{k} t_{ij} \leq 1$ $(1 \leq i \leq N, 1 \leq j \leq k)$. In our experiments, we choose $m = 2$. The EM algorithm is also modified to deal with fuzzy assignment of data points. First, during the E-step, it can be shown that minimizing Q with respect to t_{ij} leads to a quadratic optimization problem with inequality constraints. The Karush-Kuhn-Tucker (KKT) Optimality Criteria provides a necessary condition for solving the optimization problem [9]. Letting λ_i to be the Lagrange multipliers, we obtain $Q' = Q - \sum_{i=1}^{N} \lambda_i (\sum_{j=1}^{k} t_{ij} - 1)$. The t_{ij} that satisfies the following conditions are candidate solutions to the optimization problem:

$$\frac{\partial Q'}{\partial t_{ij}} = 0 \; (1 \leq j \leq k), \quad \sum_{j=1}^{k} t_{ij} \leq 1 \tag{3.6}$$

$$\lambda_i \left(\sum_{j=1}^{k} t_{ij} - 1 \right) = 0, \quad \lambda_i \leq 0 \tag{3.7}$$

Equation (3.7) can be satisfied under the following two situations:

1. If $\lambda_i = 0$, upon simplifying the results of Equation (3.6), we obtain:

$$\frac{\partial Q'}{\partial t_{ij}} = 2 t_{ij} D_{ij} - 2\beta \sum_{j=1}^{k} t_{ij} = 0 \tag{3.8}$$

where $\beta = \gamma + C_{10} - C_{01}$. Simplifying this yields:

$$t_{ij} = \frac{\beta \sum_{j=1}^{k} t_{ij}}{D_{ij}} \tag{3.9}$$

Summing up t_{ij} over all j leads to the solution $t_{ij} = 0$. In this case, the value of the objective function goes to some constant q.

2. If $\sum_{j=1}^{k} t_{ij} - 1 = 0$, then the following condition holds:

$$\frac{\partial Q'}{\partial t_{ij}} = 2 t_{ij} D_{ij} - 2\beta \sum_{j=1}^{k} t_{ij} - \lambda_i = 0 \tag{3.10}$$

So we obtain the fuzzy degree of membership for each point:

$$t_{ij} = \frac{1}{\sum_{s=1}^{k} (D_{ij} / D_{is})} \tag{3.11}$$

Nevertheless, since the objective function is not a convex function, the KKT conditions are necessary but not sufficient. We must verify that the t_{ij} obtained minimizes the overall objective function. To do this, we replace t_{ij} from Equation 3.11 into Q and compare it against the result of using $t_{ij} = 0$. If the former value is greater, then t_{ij} is re-assigned to 0 to minimize the objective function Q. Otherwise, we retain the value of t_{ij} given by Equation 3.11.

Similar to hard detection, the following equations are updated in M-step:

$$\mu_j = \frac{\sum_{i=1}^{N} t_{ij}^2 x_i}{\sum_{i=1}^{N} t_{ij}^2}$$

$$\Sigma_j = \frac{\sum_{i=1}^{N} t_{ij}^2 (x_i - \mu_j)^T (x_i - \mu_j)}{\sum_{i=1}^{N} t_{ij}^2}$$

4 Experiments

Table 1: Description of Data Sets

Data sets	# of features	% of outliers	% of labels	# of instances
Shuttle	9	4.2%	5%	3132
Optical handwritten	58	3%	5%	3547
Intrusion detection	38	9%	5%	11000
Physiological	9	1.2%	5%	4050

We have performed extensive experiments using real data sets to evaluate the performances of our proposed algorithms. The data sets used in our experiments are summarized in Table 1. The two variants proposed in this paper are denoted as HardS and FuzzyS, which correspond to hard anomaly detection and fuzzy anomaly detection respectively. The parameters of these algorithms are selected using the methodology described in our technical report [4].

We compared the performances of our algorithms against a distance-based anomaly detection algorithm called **K-dist** [6] and a density-based algorithm known as **LOF** [2]. K-dist uses the distance between a data point to its k-th nearest neighbor to be the anomaly score. LOF, on the other hand, computes the anomaly score in terms of the ratio between the density of a point to the density of its k nearest neighbors. Semi-supervised classification is another approach for detecting anomalies using labeled and unlabeled examples. We apply the semi-supervised classification algorithm (**NBEM**) developed by Nigam et al. [8], which combines naïve Bayes with EM algorithm. Finally, we also employ a supervised learning algorithm—the naïve Bayes classifier (**NB**)—as a baseline classifier for comparison purposes. Note that the naïve Bayes classifier is trained using the actual class labels for the entire data set. Therefore, we do not expect any of the unsupervised or semi-supervised algorithms to outperform **NB**.

We employ four evaluation metrics to compare the performances of our algorithms: Precision(P), Recall(R), F-measure(F), and False Alarm rate (FA).

4.1 Comparisons Among Unsupervised, Semi-supervised and Supervised Anomaly Detection Algorithms Tables 2 to 5 show the results of applying the various algorithms to four real data sets. Notice that FuzzyS and HardS algorithms generally outperform both LOF and K-dist, except on the Physiological data. In the Shuttle data, the F-measure for FuzzyS and HardS algorithms are even better than the naïve Bayes classifier. For the optical handwritten data, the unsupervised learning algorithms perform poorly because of the high-dimensional and sparse nature of the data, a situation in which the notion of distance is likely to break down.

Table 2: Shuttle($k = 2$,$\gamma = 20$)

	FuzzyS	HardS	K_dist	LOF	NBEM	NB
R	0.9103	0.7374	0.3636	0.1212	0.8186	0.935
P	1.0000	1.0000	0.3636	0.1212	0.0682	0.754
F	0.9531	0.8489	0.3636	0.1212	0.1259	0.835
FA	0.0043	0.0157	0.0280	0.0387	0.0007	0.0053

Table 3: Optical handwritten data($k = 5$,$\gamma = 85$)

	FuzzyS	HardS	K_dist	LOF	NBEM	NB
R	0.0810	0.7111	0.05	0.05	1.0000	0.8
P	0.8250	0.8000	0.05	0.05	0.9750	0.923
F	0.1903	0.7529	0.05	0.05	0.9873	0.857
FA	0.1376	0.0067	0.01	0.01	0.0000	0.001

Table 4: Intrusion detection data($k = 2$,$\gamma = 50$)

	FuzzyS	HardS	K_dist	LOF	NBEM	NB
R	0.4011	0.7213	0.5123	0.4322	0.8336	0.988
P	0.9980	0.9940	0.3333	0.1167	0.9520	1.0000
F	0.5722	0.8360	0.4039	0.1838	0.8889	0.994
FA	0.0745	0.0192	0.0439	0.0402	0.0095	0.0006

Table 5: Physiological data($k = 2$,$\gamma = 50$)

	FuzzyS	HardS	K_dist	LOF	NBEM	NB
R	0.8824	0.9167	0.92	0.060	1.0000	0.875
P	0.9000	0.8800	0.92	0.060	0.0400	1.0000
F	0.8911	0.8980	0.92	0.060	0.0767	0.933
FA	0.0015	0.0010	0.002	0.0235	0.0000	0.0000

Both semi-supervised classification and semi-supervised anomaly detection algorithms employ labeled data to improve the detection of anomalies. Our empirical results show that HardS and FuzzyS algorithms perform better than NBEM on two of the four data sets. To understand why one algorithm may be better than the other, Figure 2 shows a comparison between our proposed algorithms and NBEM on the Physiological data set when the percentage of outliers is varied from 1% to 35%. Our experimental results suggest that FuzzyS and HardS tend to outperform NBEM when the percentage of outliers is small, whereas NBEM performs better when the percentage of outliers is more than 20%. We expect semi-supervised classification algorithms to do a poor job when the distribution is highly skewed because estimating the probability distribution becomes a challenging task if there are very few examples in the outlier class. It should be noted that in anomaly detection, the scenario where anomalies account for more than 20% of the data is a very rare situation indeed. In short, semi-supervised anomaly detection would be a better choice than semi-supervised classification when the percentage of outliers is small.

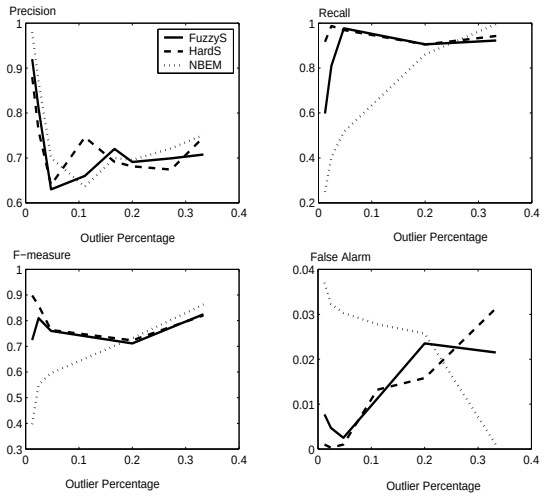

Figure 2: Performances while varying the percentage of outliers

4.2 The Utility of Labeled Information Figure 3 shows the effect of varying the amount of labeled data while keeping the amount of unlabeled data to be the same. We use the FuzzyS anomaly detection algorithm on the Intrusion Detection data set for this experiment. Naturally, having more labeled data helps to increase the detection rate and to reduce the false alarm rate. When the percentage of labeled data increases, the detection rate and false alarm rate for FuzzyS both improve. Furthermore, notice that our algorithm achieves a very high detection rate (more than 99%) and low false alarm rate (less than 2%) when the percentage of labeled data is more than 10%.

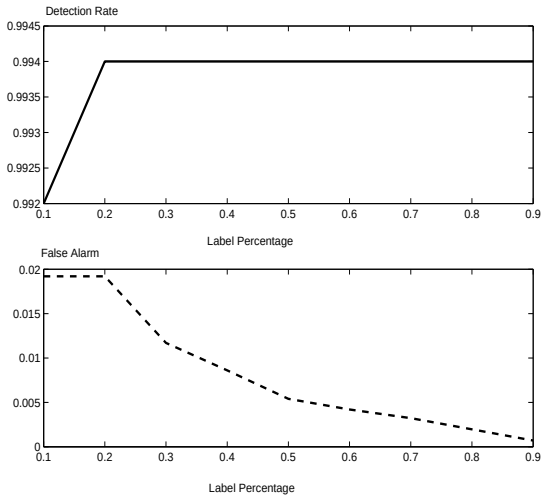

Figure 3: Detection rate while varying the percentage of labeled data

5 Conclusion

This paper introduces a probability-based framework for detecting anomalies using labeled and unlabeled data. The labeled examples are used to guide the anomaly detection algorithm towards distinguishing data points that are hard to classify (e.g., anomalies that are located in high density regions or normal points that are located in low density regions). Experimental results using real data sets confirmed that our proposed algorithm is generally more superior than unsupervised anomaly detection algorithms such as k-dist and LOF. We also compared the performances of proposed HardS and FuzzyS algorithms against a semi-supervised classification algorithm called NBEM. The results show that semi-supervised anomaly detection tends to do better when the percentage of outliers is considerably smaller than the percentage of normal points.

References

[1] S. Basu, M. Bilenko, and R. J. Mooney. A probabilistic framework for semi-supervised clustering. In *KDD '04: Proc. of the 2004 ACM SIGKDD Int'l Conf. on Knowledge discovery and data mining*, pages 59–68, 2004.

[2] M. M. Breunig, H.-P. Kriegel, R. T. Ng, and J. Sander. Lof: identifying density-based local outliers. In *SIGMOD '00: Proc. of the 2000 ACM SIGMOD Int'l Conf. on Management of data*, pages 93–104, 2000.

[3] E. Eskin. Anomaly detection over noisy data using learned probability distributions. In *ICML '00: Proc. of the 17th Int'l Conf. on Machine Learning*, pages 255–262, 2000.

[4] J. Gao, H. Cheng, and P. Tan. A probability based anomaly detection algorithm with labeled examples. *Michigan State University, Technical Report*, 2005.

[5] J.C.Bezdek. *Pattern Recognition With Fuzzy Objective Function Algorithms*. Plenum Press, New York, 1981.

[6] W. Jin, A. K. H. Tung, and J. Han. Mining top-n local outliers in large databases. In *KDD '01: Proc. of the Seventh ACM SIGKDD Int'l Conf. on Knowledge discovery and data mining*, pages 293–298, 2001.

[7] T. Mitchell. *Machine Learning*. McGraw Hill, 1997.

[8] K. Nigam, A. K. McCallum, S. Thrun, and T. M. Mitchell. Text classification from labeled and unlabeled documents using em. *Machine Learning*, 39(2/3):103–134, 2000.

[9] R.Fletcher. *Practical methods of optimization*. John Wiley & Sons, Chichester, second edition, 1986.